Applications Note

SEBINI: Software Environment for Biological Network Inference

Ronald C. Taylor¹*, Anuj Shah¹, Charles Treatman², and Meridith Blevins³

¹Computational Biology & Bioinformatics Group, Pacific Northwest National Laboratory, Richland, Washington, USA, ²Oberlin College, Oberlin, Ohio, USA, and ³Case Western Reserve University, Cleveland, Ohio, USA

ABSTRACT

Summary: The Software Environment for Biological Network Inference (SEBINI) has been created to provide an interactive environment for the deployment and evaluation of algorithms used to reconstruct the structure of biological regulatory and interaction networks. SEBINI can be used to compare and train network inference methods on artificial networks and simulated gene expression perturbation data. It also allows the analysis within the same framework of experimental high-throughput expression data using the suite of (trained) inference methods; hence SEBINI should be useful to software developers wishing to evaluate, compare, refine, or combine inference techniques, and to bioinformaticians analyzing experimental data. SEBINI provides a platform that aids in more accurate reconstruction of biological networks, with less effort, in less time.

Availability: A demonstration web site is located at https://www.emsl.pnl.gov/NIT/NIT.html. The Java source code and PostgreSQL database schema are available freely for non-commercial use.

Contact: ronald.taylor@pnl.gov

1 INTRODUCTION

Reconstruction of regulatory and signaling networks is a critical task in systems biology. High-throughput molecular biology experiments are now producing mRNA expression data in quantities large enough for researchers to attempt to reconstruct the structure of gene transcription networks based primarily on state correlation measurements. Protein expression and activation measurements are soon to follow, allowing similar work on protein-protein interaction and signaling networks.

Researchers have previously created artificial networks and simulated expression data to test their specific network inference algorithm development (Ideker et al., 2000; Tamada et al., 2003; Zak et al., 2001). A collection of static data sets for four network topology types, with corresponding synthetic gene expression data, has been made available for download as flat files (Mendes et al., 2003). The ASIAN web site is available where users can infer relationships from an expression data set to a binned expression data set, original expression data. Note that we have one-to-many relationships among possibly huge data sets, allowing easy drill down from a network set to a specific network to a specific node or edge, and (2) producing an efficient, understandable mapping from the inferred networks and inferred edges back to the corresponding original expression data. Note that we have one-to-many relationships from an expression data set to a binned expression data set, of-the-art algorithms to experimentally generated high-throughput data (van Someren et al., 2002). The “Software Environment for Biological Network Inference” (SEBINI) project at the Pacific Northwest National Laboratory (PNNL) is designed to fill this gap, as an aid in the reconstruction of the structure; that is, the “wiring diagram” of mRNA and protein networks. SEBINI provides a web-based environment that allows inference algorithms to be compared, trained, refined, and then employed on experimental data.

We assume that users of SEBINI seek to directly infer genetic regulatory networks from high-throughput microarray mRNA expression data, and protein interaction and signaling networks from high-throughput quantitative protein data. Briefly, methods using high-throughput data rely on searching for patterns of partial correlation or conditional probabilities that indicate causal influence (Sprites et al., 2000). Such patterns of partial correlations found in the high-throughput data, possibly combined with other supplemental data on the genes or proteins in the proposed networks, or combined other information on the organism, are the basis upon which the algorithms in SEBINI’s toolkit infer networks. In other words: SEBINI may be useful in inferring the topology of any network where the change in state of one node can affect the state of other nodes.

2 SEBINI ARCHITECTURE

SEBINI uses a standard three-tier architecture: (1) a web-based client user interface, (2) an application logic middle tier consisting of a suite of Java servlets and other Java programs (more than 100 Java classes), and (3) a relational database storing the data required by the middle tier. Inferred networks (as well as the raw data, discretized data, and algorithm parameter selections used to generate the networks) are permanently stored in the database for visualization, topological and statistical analysis, and for later export in a human-readable or program-specific format. Inference and discretization (binning) algorithms can be any sort of executable program; a Java handler class is added for each new algorithm to handle communication between the invocation web page, the database, and the algorithm. Security is implemented on a project basis, with one owner and possibly multiple users per project.

Major design issues included (1) the interface for user navigation among possibly huge data sets, allowing easy drill down from a network set to a specific network to a specific node or edge, and (2) producing an efficient, understandable mapping from the inferred networks and inferred edges back to the corresponding original expression data. Note that we have one-to-many relationships from an expression data set to a binned expression data set,

*To whom correspondence should be addressed.
as well as a one-to-many relationship between a binned data set and the inferred network and inferred edges created by the selected inference algorithm. Records for each of these data types are permanently stored and connected to the appropriate records of the other data types. Other design decisions: all inter-servlet communication is routed through a CentralControl servlet, for a clear (and reusable) flow of control. Each binning and inference algorithm is invoked in a separate Java thread that performs job posting to the database, thus allowing dynamic monitoring of job progress by the user. Jobs are timed to the millisecond, allowing comparison between algorithms of relative speed versus relative power.

SEBINI was initially implemented on a Dell desktop running Red Hat Linux, using Java ver. 1.4, PostgreSQL ver. 7.4, and Tomcat 4.1. SEBINI has also been installed on a Windows 2003 Web Server. Machine-specific parameters are stored in an easily changed properties text file. Mathworks’ MATLAB is required for some of the inference algorithms.

3 SEBINI CAPABILITIES

Capabilities of interest to all users include:
1. Upload of several types of experimental data for input into selected binning and network inference algorithms.
2. Several choices of inference methods from the growing toolkit, currently including algorithms from classical statistics (e.g., Pearson correlation, as a baseline), Bayesian networks (e.g., Hartemink et al., 2005, and Sachs et al., 2005), and information theory (mutual information-based; e.g., Margolin et al., 2006).
3. Inference results (inferred networks) that can be permanently stored and further analyzed. For each network, the user can view a summary page, a topological characteristics and statistics page, a graph visualization using Cytoscape (Shannon et al., 2003) invoked via Java Web Start, summary pages for each node and edge showing the raw and binned node states, and job pages that record how the binning and inference tasks proceeded.
4. Direct comparison of network inference methods on common synthetic or experimental data sets.
5. A planning tool for experiments. How well can one do in reconstructing the edges in a genetic transcriptional network, given a set of expression data and a network of a given topology, using a given inference method? SEBINI will allow predictions, using different inference methods, on what can be reconstructed of the topology (regulatory connections) of a network of a given size and complexity.
6. Export of inferred network structures as input to other tools (e.g., for dynamical modeling) and export of human-readable reports on the networks, with various topological characteristics noted.
7. To support algorithm developers, SEBINI also allows:
   a. Artificial data sets (e.g., topologies, perturbations, and node input functions) that can be dynamically created and stored.
   b. Dynamic, step-wise refinement of inference methods, based on results. Scoring measures (recall, precision, F-measure) are used to score performance against the simulated networks with known structure.
   c. Well-defined addition of new inference algorithms, binning techniques, and import and export methods.

10. Supervised or unsupervised training of inference methods, with supervised inference results scored against the known network topologies.
11. A guide for the interpretation of the scores produced by an inference technique. SEBINI can produce scoring distributions for a given inference method against known networks. Such distributions could then be used to determine appropriate cutoff scores for determination of the existence of a regulatory influence (an edge) to a target gene.

4 FUTURE WORK

We will continue to add to the capabilities of SEBINI: additional inference and binning algorithms, methods of generating simulated data sets, and import/export techniques. We are actively seeking algorithm developers as collaborators. We are exploring (1) refining or combining algorithms for improved results, (2) adding genome-specific annotation, (3) various types of post-processing to remove incorrect or indirect edges, and (4) the use of SEBINI as a platform for automated fitting of logic functions (e.g., Istrail and Davidson, 2005) and state equations to the state sets tied to the source and target nodes for each inferred edge in SEBINI’s database, as an additional step toward dynamic modeling.

ACKNOWLEDGMENTS

This work has been supported by the U.S. Department of Energy (DOE) through the Biomolecular Systems Initiative at PNNL, and also through PNNL’s William R. Wiley Environmental Molecular Science Laboratory (EMSL) and the EMSL Grand Challenge in Membrane Biology project, via PNNL’s Laboratory Directed Research and Development Program. PNNL is operated by Battelle for the DOE under contract DE-AC05-76RL01830. C.T. and M.B. were supported by the DOE Science Undergraduate Laboratory Internship (SULI) program.

Conflict of Interest: none declared

REFERENCES